

# Professional Software Engineering Practices in a Small but Emerging Country: Do Those Differ from Other Countries?

Dummy abstract

## ACM Reference Format:

. 2020. Professional Software Engineering Practices in a Small but Emerging Country: Do Those Differ from Other Countries?. *ACM Trans. Manag. Inform. Syst.* 1, 1 (October 2020), 27 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

The term software engineering first appeared in the late 1960s and was introduced by Bauer to describe ways to develop, manage and maintain software so that the resulting products are reliable, correct, efficient and flexible. Some 15 years later, Zerkowicz et al. [?] performed an in-depth survey of 30 companies in which they tried to establish the current state of practice in the software production industry. Their survey revealed that at that time practice was around 10 years behind on software engineering research. Almost 20 years later, in 2003 to be precise, Dr. Reifer observed that industry is a little behind academia, but industry has the capacity to close the gap very quickly [?].

A software engineer applies engineering practices to software development, and typically handles the overall system design of the software application. Software engineering is divided into several specializations, which focus on a particular software development area. A few of the most common software engineering specialties include requirements engineering, software engineering processes, software design, tools and techniques. These specialties guide software engineering practice.

Software development varies from region to region. A certain methodology may become more efficient in some region which is not considered efficient in another regions. Though most of the factors behind software development are same all the time, there are certain points which vary from region to region. For software development professionals, deciding on processes, practices and techniques is critical. Sometimes, these decisions might be based on marketing and literature bias that supports new or industry-supported practice.

Many studies concerning the practices of the software development have been conducted. Those studies convey similarities, but also differences of diverse regions, countries and communities. Previous empirical studies have gone into how software development practices were conducted in the North America, Europe, Turkey and New Zealand but to the best of our knowledge, none were carried out in Bangladesh. Therefore, this study has been conducted to focus on the current practices of software development process and to provide perception of the software industries in Bangladesh. On top of that, the aspiration of this study is to look over the methodology of software development practices, the adoption of technologies and tools by professionals, and the use of testing and deployment practices. Specifically, we are interested in the developers' and managers'

---

Author's address:

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2020 Association for Computing Machinery.

2158-656X/2020/10-ART \$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

opinion about several questions regarding software development related practices. For example, i) *What software methodology are used in your project?* ii) *Which implementation technologies and tools are adopted by software development professionals?* iii) *What type of testing and deployment practices are used?* iv) *How are the security and performance is ensured in a product of a company?* and v) *What are the personal expectations and desire reigning in the industry?*

Our study finds some similarities in development process of Bangladesh with other countries and also some interesting differences. For example, agile method is mostly used in Bangladesh but scrum has greater usage in New Zealand. On the contrary, textual description is used for requirements gathering in both Bangladesh in Netherlands. A lot number of developers of Bangladesh use frameworks for ensuring security and cloud services for scalability.

The rest of the paper is structured as follows. Section 2 presents previous related studies. Section 3 describes the demographics of the respondents and their projects. Section 4 presents the results obtained from this study and finally, Section 5 concludes the paper.

## 2 RELATED WORKS

There are many works about software development procedures outside of our country, and those researches were based on that particular country. Nevertheless, we have come to know a great extent of knowledge from those researches.

In 2012, Vonken et al. [?] conducted a survey among Dutch software producing organizations to determine whether there is a gap between the current state of the practice and state of the art in software engineering. From 99 respondents, they extracted 22 interesting observations. These observations mark insights into the development process that they found unusual, or surprising, at least from an academic point of view. This unusualness could either stem from certain principles being applied less or more frequently than one would expect, or from unexpected correlations that were observed between factors.

The survey in [?] studies the software practices in Turkey to characterize and understand the state of its SE practices. The military and defense software sectors are quite prominent in Turkey, especially in the capital Ankara region, and many SE practitioners work for those companies. 54% of the participants reported not using any software size measurement methods, while 33% mentioned that they have measured lines of code (LOC). In terms of effort, after the development phase (on average, 31% of overall project effort), software testing, requirements, design and maintenance phases come next and have similar average values (14%, 12%, 12% and 11% respectively). Respondents experience the most challenge in the requirements phase. Waterfall, as a rather old but still widely used life-cycle model, is the model that more than half of the respondents (53%) use. The next most preferred life-cycle models are incremental and Agile/lean development models with usage rates of 38% and 34%, respectively. The Waterfall and Agile methodologies have slight negative correlations, denoting that if one is used in a company, the other will less likely to be used.

A recent survey conducted by Wang et. al. [?] in 2018 shows that, New Zealand professionals use similar methodologies as professionals in other countries. Popular programming languages differ somewhat to popular languages in other rankings. Quality assurance is rather ad-hoc and the release process is inspired by agile software development principles. Our findings highlight some differences of the New Zealand software industry to other countries. Furthermore, we identified some strengths and weaknesses related to processes and practices. Our findings can help software professionals and organizations reflect on (and potentially adjust) the way they work.

From another study [?], we report on the software development techniques used in the New Zealand software industry, paying particular attention to requirements gathering. We surveyed a selection of software companies with a general questionnaire and then conducted in-depth interviews with four companies. Our results show a wide variety in the kinds of companies

undertaking software development, employing a wide range of software development techniques. Although our data are not sufficiently detailed to draw statistically significant conclusions, it appears that larger software development groups typically have more well-defined software development processes, spend proportionally more time on requirements gathering, and follow more rigorous testing regimes.

The study conducted by Sison et. al. [?] presents exploratory survey and case study results on software practices of some software firms in five ASEAN countries (Malaysia, Philippines, Singapore, Thailand and Vietnam), and provides directions for further research on software practices in the ASEAN/Southeast Asian region.

### 3 DEMOGRAPHIC

#### 3.1 Respondent's Age

Age of a person often determines his/her knowledge and experience with the focus of the survey. When administering a survey about SE industries, a respondent in his 24s and 25s will most likely answer the question differently than a respondent his 35s. The age distribution of our survey respondents is shown in fig. 1.

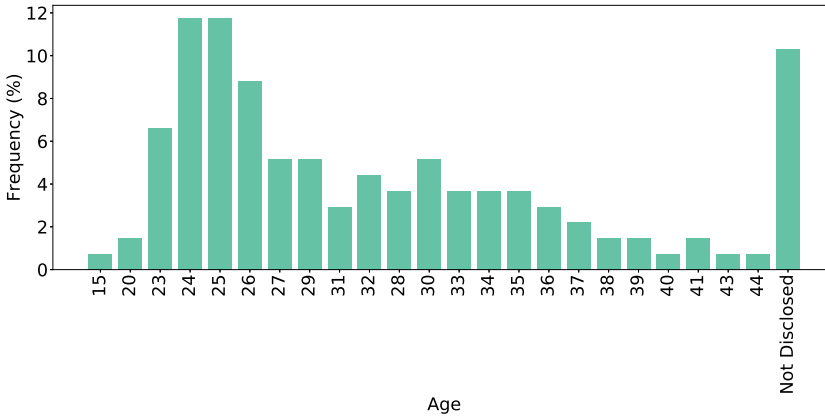


Fig. 1. Respondent's Age

#### 3.2 Work Experience

Around 34% of our respondent's has less than 2 years software development experiences. As the fig. 2 shows, we have respondent's with various years of work experience which will have positive impact by accumulating valuable results from wider audience base.

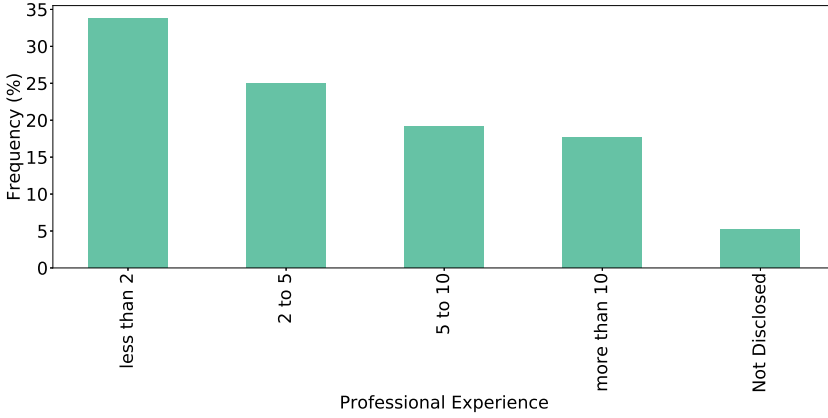


Fig. 2. Work Experience

### 3.3 Respondent's Current Role

From the fig. 3 we see most of the respondents (70%) were software developers and approximately 18% of our respondent's are working at manager level post which have appreciable impact on getting expectations from managers.

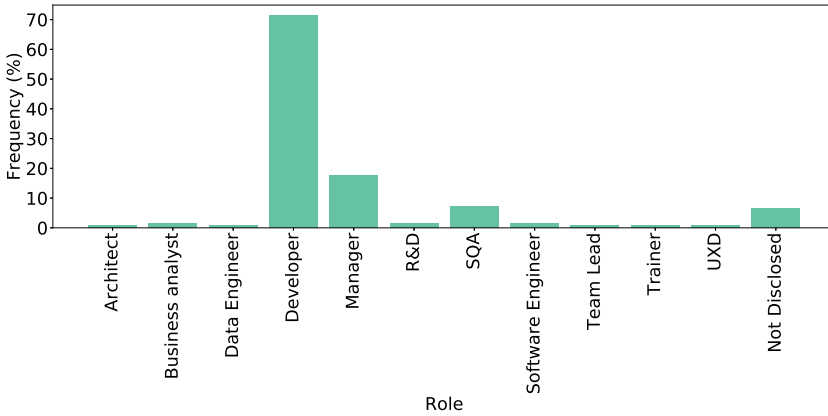


Fig. 3. Respondent's Current Role

## 4 RESULTS

### 4.1 RQ1: What software methodology are used in your project?

Software development methodology in software engineering is a framework that is used to pursue the development of information systems in a very deliberate, structured and methodical way. To find out the overall answer of this question, we report the following results:

- Software development methodologies (Q 6).
- Requirements gathering (Q 7).
- Most time consuming software development activities (Q 8).

**4.1.1 Software development methodologies.** The most-widely used model is Agile with usage rate of 64%. The next widely-used model is scrum with usage rates of 46%. The other methodologies has lower usage rates, namely: pair programming (20%), Waterfall (12%) etc. From the study we see that the most acceptable model that was regularly and always used is the agile model in Bangladesh but the usage of the scrum (44%) in New Zealand has greater usage followed by agile (30%) [?] and in Turkey, waterfall is mostly used based on the earlier 2015 survey [?]. Again, in both Bangladesh and New Zeland, extreme programming (XP) has a lower percentage of usage.

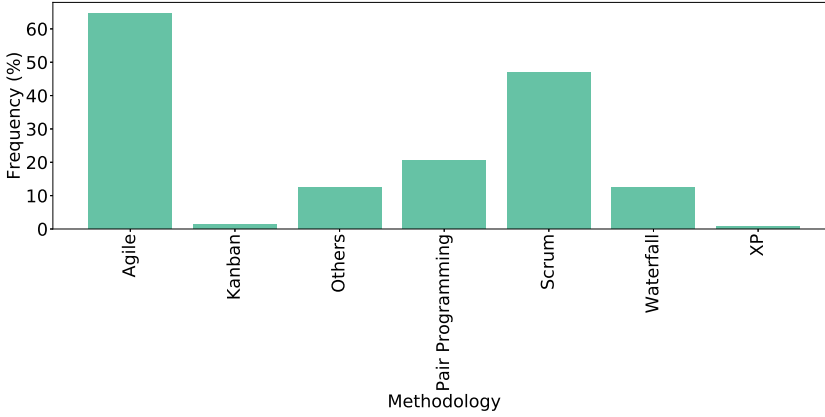


Fig. 4. Software development methodologies

**4.1.2 Requirements Gathering.** The most critical activity that always arises during software development is the collecting requirement of the proposed system. According to fig. 5, using plain text (44%) and story board (41%) are the most widely used requirements gathering. This result is similar with the survey of Vonken et al. [?]. From their study we can find that the textual description of specifying requirements is a firm favourite in Netherlands. The other requirements gathering usage rates are: Use case (36%), GUI prototype (35%), grooming session (30%) etc. This is an important finding that requires further analysis for causes and to analyze the potential effects of not documenting requirements.

**4.1.3 Development activities timeline.** In this section participants were asked about the most time consuming software developing activities they had spend. As we see in fig. 6, most of the time spent in implementation stage according to 65% our respondents and requirement analysis stage requires second most according to 45% response. The other usages are: Program design (37%), project planning (30%), testing (19%), maintenance (17%) etc. According to study [?], most time was spent on implementation and coding and relatively less time was spent on maintenance in Bangladesh and New Zealand. But requirement analysis, the activity, requires the second most time to spend in Bangladesh where in New Zealand, it is testing (36%) practices.

## 4.2 RQ2: Which implementation technologies and tools are adopted by software development professionals?

Our survey included five questions to find technologies and tools that are adopted by software development professionals. To answer this question fully, we report the following results:

- Technology Platform (Q 9).
- Operating System (Q 10).

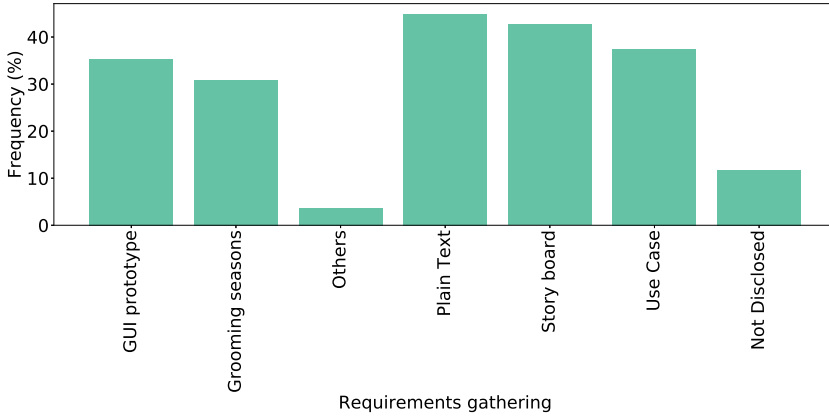


Fig. 5. Requirements gathering

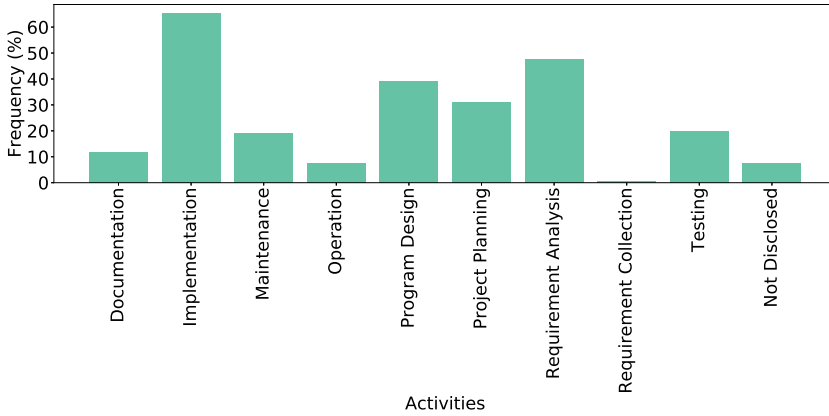


Fig. 6. Software development activities

- Programming Language (Q 11).
- Framework (Q 12).
- IDE (Q 13).

**4.2.1 Technology Platforms.** Participants were allowed to choose multiple options. As shown in fig. 7, most of the respondents (80%) worked in web platform. The rests were mobile (45%), Desktop (30%), Embedded/IOT (8%). Most respondents develop products and services for web platforms in both Bangladesh and New Zealand based on [?].

**4.2.2 Operating Systems.** Most of our respondent's used linux based operating system (56%). The second best used operating system is windows (45%). Windows is mostly used among developers of New Zealand based on the survey [?], but linux is mostly used in the case for Bangladeshi developers.

**4.2.3 Programming Languages.** According to fig. 9, around 61% of our respondent's used Java and Javascript each. Other languages like php (25%), python (25%), c# (18%) are also used which indicates that the software engineers are not inclined towards a single specific language. Also

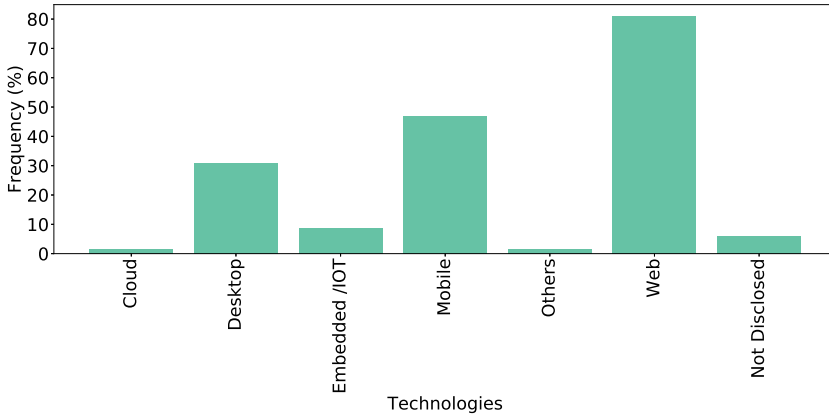


Fig. 7. Technology Platforms

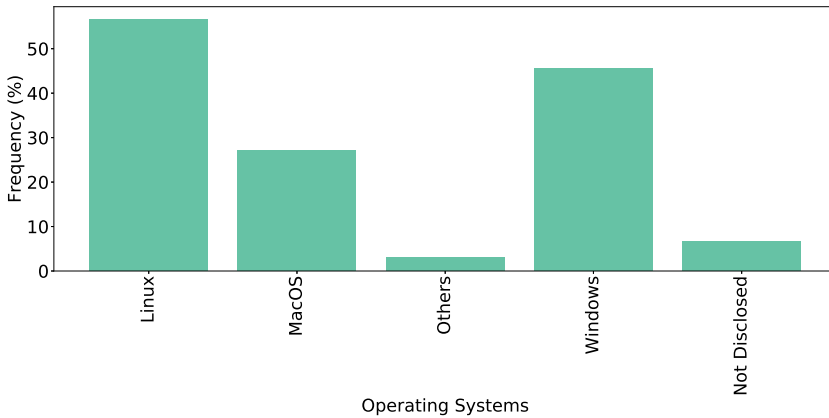


Fig. 8. Operating Systems

the choice of programming languages used for development can have important inferences for the testing practices of a software company. Based on [?], Java ranks quite low in New Zealand where it is the second most used in Bangladesh and the mostly used language in Turkey [?]. Again, python did not get a respective place in the rank of used languages in New Zealand but it is used significantly in Bangladesh.

**4.2.4 Frameworks used in development.** As shown in fig. 10, variety of frameworks have been used during development. Spring boot (37%) is the mostly used framework in the industry. ASP.NET, Django and Laravel are used in same proportion (14%).

**4.2.5 IDE's used by the respondent's.** According to fig. 11, IntelliJ, a Java integrated development environment for developing computer software for enterprise, mobile, and web development used by most of the respondents (43%). The other IDEs used in SE industries are: visual studio (30%), Eclipse (24%), PyCharm (16%), NetBeans (10%), Android Studio (6%).

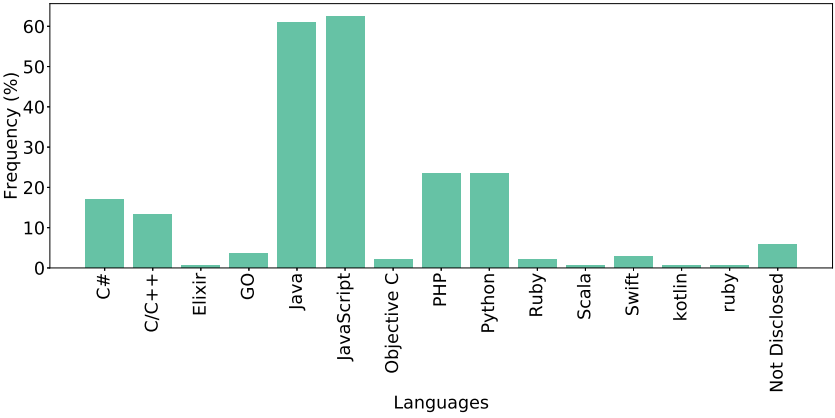


Fig. 9. Languages used in software development

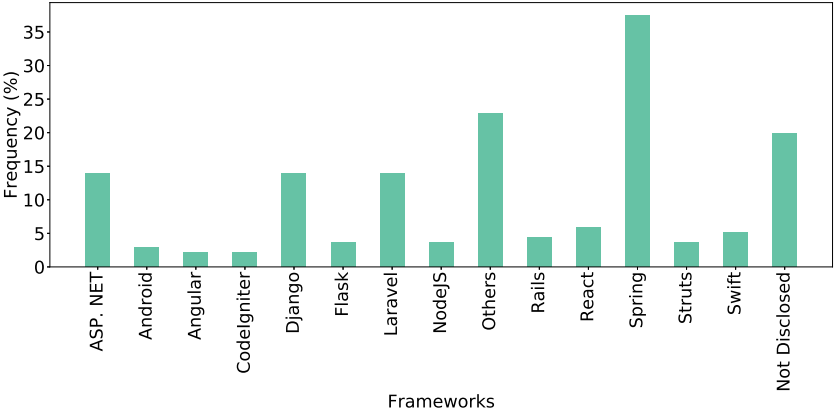


Fig. 10. Frameworks

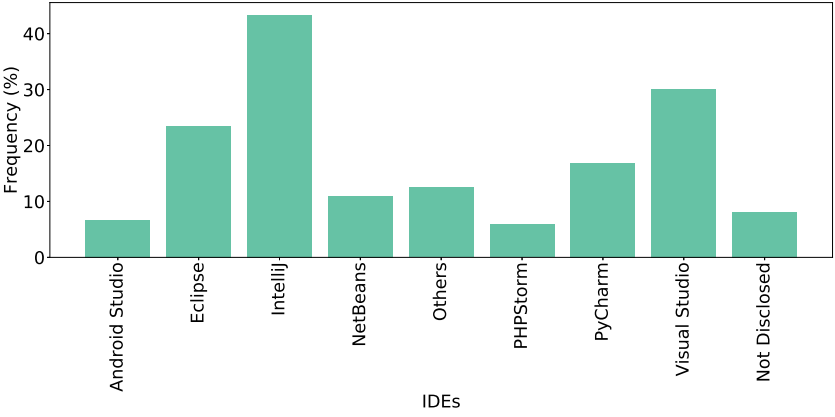


Fig. 11. IDE's



### 4.3 RQ3: What type of testing and deployment practices are used?

Testing is an important process in improving the quality of the software product. The purpose of this process is to find errors, which might occur during specification, design and code generation. Here we report the following results next:

- Software Testing Practices (Q 14)
- Level of Automated Testing (Q 15)
- Tools Used in Testing and QA (Q 16)
- Continuous Deployment tools (Q 17)
- Version Control (Q 18)

**4.3.1 Software Testing Practices.** According to fig. 12, several number of testing practices are used during software development. The results show most of the organizations carried out unit testing (53%), functional testing (49%), functional testing (49%), user acceptance testing (39%), GUI testing (31%) etc.

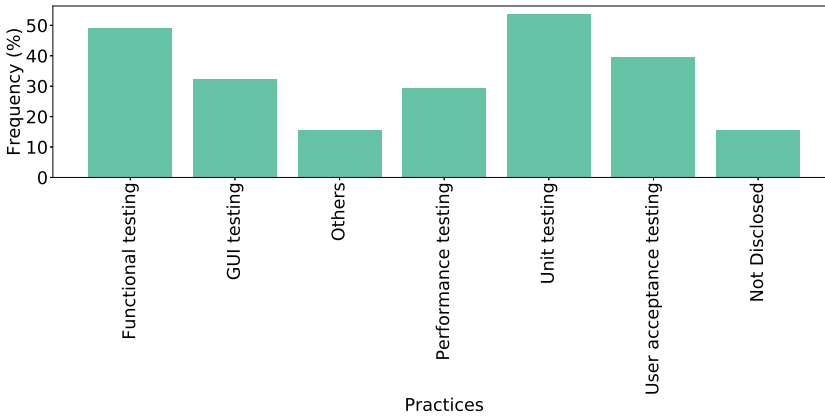


Fig. 12. Testing Practices

**4.3.2 Level of Automated Testing.** Question 15 asked about the level of automated testing. The responses were gathered using the Likert scale. This denotes that different respondents have very different practices in this context, i.e., some heavily practice automated testing, while others favor manual testing. Results are shown in fig. 13.

**4.3.3 Tools Used in Testing and QA.** Q 16 asked about the tools used in testing and quality assurance. According to fig. 14, we see that most of the respondents have been used XUnit( eg, JUnit, NUnit) (30%), selenium (27%), Jenkins (20%), others (9%).

**4.3.4 Deployment Tools.** According to fig. 15, we see that most of the respondents use AWS code-deploy (12%) and JenKins (12%). The other deployment tools are Bamboo (5%), teamcity (4%), octopus (2%). Respondents voted none (4%) as they didn't use any deployment tools.

#### 4.3.5 Version Control.

Respondents were allowed to select more than one option. As shown in figure 16, Git (78%) and BitBucket (29%) are mostly-used version control in the software industry. Beside these SVN (5%), others (4%) are used. The 2018 Stack overflow survey[?] reports that the most popular version control system is Git (87.2% developer uses Git) and the second most popular is SVN (16.1% developer uses SVN). However, in our survey, we found slightly different result, the most popular version

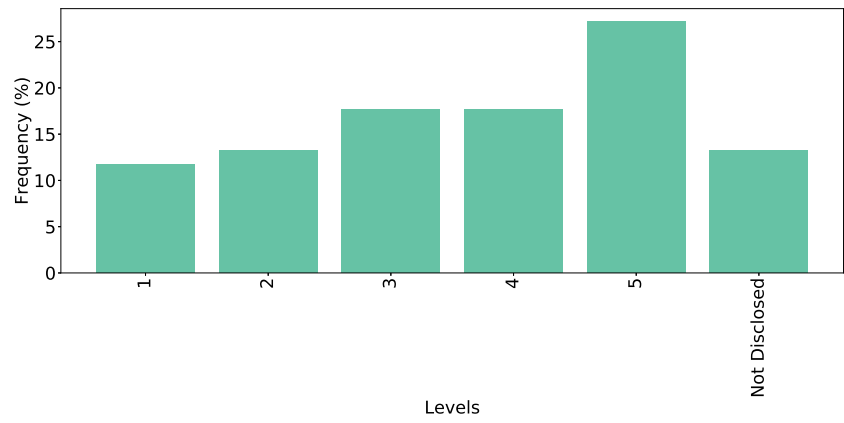


Fig. 13. Automated Testing Level

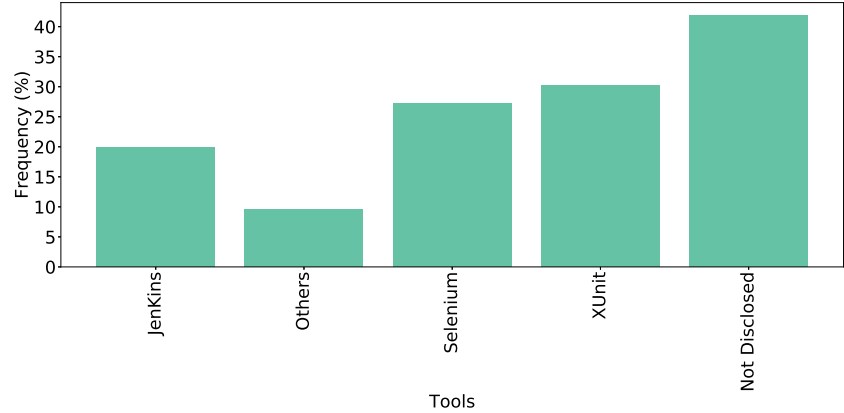


Fig. 14. Testing & QA Tools

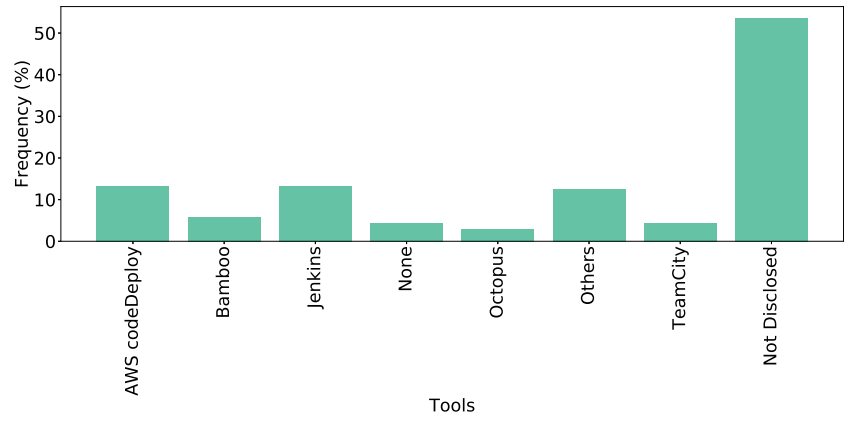


Fig. 15. Deployment Tools

control system is Git and the second most popular is Bitbucket. This might be related to the declining popularity of SVN over the years. From the Stack overflow survey over the range 2017-2019, it is clear that SVN is losing popularity to Git. Nowadays, SVN is mainly used for versioning legacy projects. As the SE industry of Bangladesh is quite new, we observe such discrepancy in the popularity of SVN.

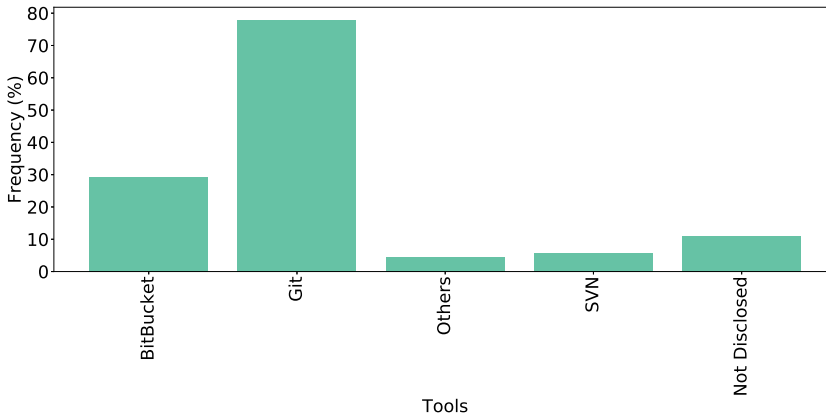


Fig. 16. Version Control

#### 4.4 RQ4:How are the security and performance is ensured in a product of a company?

This particular question covers how a company secures its code from external attack and maintain its performance after deployment. It also slightly deals with the problem of scalability. Performance depends greatly on whether a product has scalability or not. So, we will cover the following points here:

- Security
- Performance
- Scalability

##### 4.4.1 Security.

Security refers to confidentiality, integrity and authenticity. To maintain security of software product SE industry of Bangladesh is mainly dependent on Standard process which includes cloud based security, third party software, OS hardening followed by uses of security tools, network level measures and framework security. As found by Harrison et al.[?] network level measures are the first choice in ensuring security of software product which matches with our findings. Srinivasan et al.[?] listed top 10 web framework in terms of security testing where Spring framework achieved 7<sup>th</sup> position. From figure 10 we have seen that most of our respondents use Spring framework for software development. This might be the reason that a lot number of respondents dependent on framework for ensuring security.

- (1) **Application Side Measures:** Security is ensured at the software or application side. 2.19% respondents implemented security measures at application level.

*At application level, could not achieve others yet. [#72]*

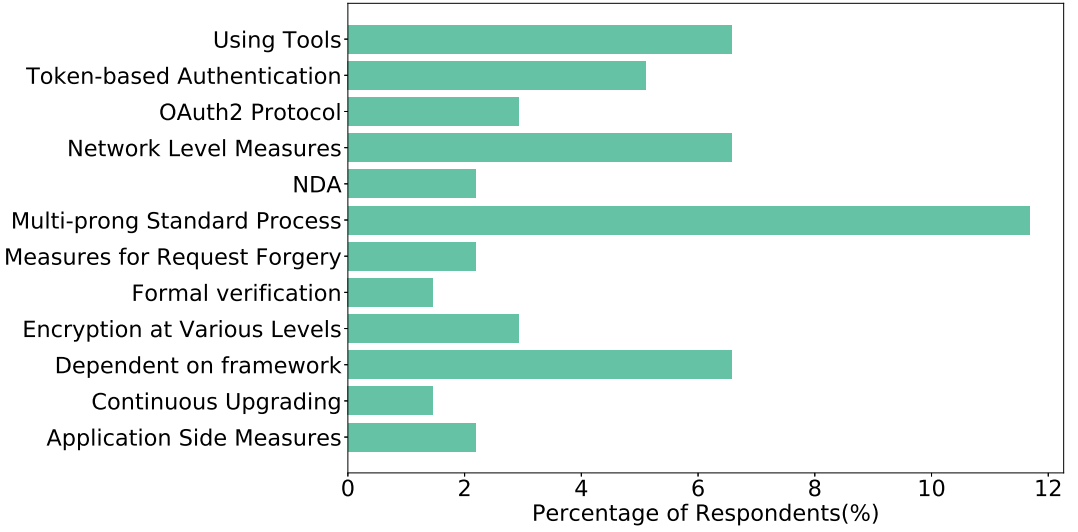


Fig. 17. Measures to maintain security of products

- (2) **Server Side Measures:** Security is ensured at the server side. 0.73% respondents implemented security measures at server level.

*We use github/bitbucket repository systems to code maintain and store. We use server side security on AWS and other third party server host.. [#48]*

- (3) **Network level Measures:** Network level measures include IP-white-listing, port-blocking, VPN and use of HTTPS in software. 6.57% respondents use at least one of the before mentioned strategy to ensure security.

*network blocking and common security measures. [#2]*

- (4) **Formal Verification:** Security threats can be mitigated by a code level review . 1.46% respondents ensured the practice of formal code review

*There are some basic guidelines that we must follow and while code review this needs to be an absolute part that needs to be checked before the code gets merged. [#112]*

- (5) **Measures for request forgery:** Measures against cross site forgery is implemented to ensure security. This includes CORS or cross origin resource sharing, CSRF or cross site request forgery or one click attack or XSSRF. 2.19% respondents implemented measures against Cross-site request forgery to ensure security.

*Security testings like: SQL injection, cross-site scripting, CSRF, API security, use of https, detecting malicious / suspicious HTTP requests and auto-blocking. [#42]*

- (6) **OAuth 2.0** : OAuth 2.0 is the industry-standard protocol for authorization. OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications, mobile phones, and living room devices. Many (2.92%) respondents use OAuth 2.0 protocol as the main way of maintaining security.

*OAuth 2.0, JWT, Token Base Authentication, CORS Filter, XSRF. [#127]*

- (7) **Token-based authentication** : A token-based authentication system allows users to enter their username and password in order to obtain a token which allows them to fetch a specific resource - without using their username and password. Once their token has been obtained, the user can offer the token which offers access to a specific resource for a time period - to the remote site. 5.11% people expressed its eligibility.

*Token based authentication for all of my rest service. [#80]*

- (8) **Multi-prong Standard Process** : Following multiprong standard processes is the best way to enforce security. About 10.22% of the total respondents expressed this opinion.

*Multi Prong Standard Processes and Products. [#110]*

- (9) **Encryption at Various Levels** : Not only at password, but also in every level of software, encryption is first and foremost to ensure its security.

*use encryption at different level of software (server, network, transmission layer, database and software layer.). [#35]*

- (10) **Use of tools** : Respondents use various open-source/ paid tools for scanning and testing. These tools help to find security threats in an existing system. The tools includes OWASP based tools and penetration testing tools

*use encryption at different level of software (server, network, transmission layer, database and software layer.). [#35]*

- (11) **Dependent on Framework** : Common framework provides basic to intermediate level security measures in the application. 6.57% respondents depends on framework for security. The framework includes spring and HDIV.

*https, popular framework which already prevents some type of attacks. rest of the things on case-by-case basis). [#79]*

- (12) **Non-disclosure agreement** : 2.19% refused to provide any details due to non-disclosure agreement.

*As we are regulated by Bangladesh Bank as a payment processing platform, we need to maintain certain security compliance which cannot be disclosed here. The security compliance is similar to the PCI DSS compliance which I can say. [#10]*

- (13) **Continuous Upgrading:** As security threats are evolving thus current systems need continuous upgradation in a frequent interval. 1.46% respondents said they arrange frequent hackathon, workshop, security audits to address the continuously evolving security threat.

*We run security audit of our office environment. We also conduct security session per 6 months to introduce latest trend in threats and what we can do to avoid it. [#57]*

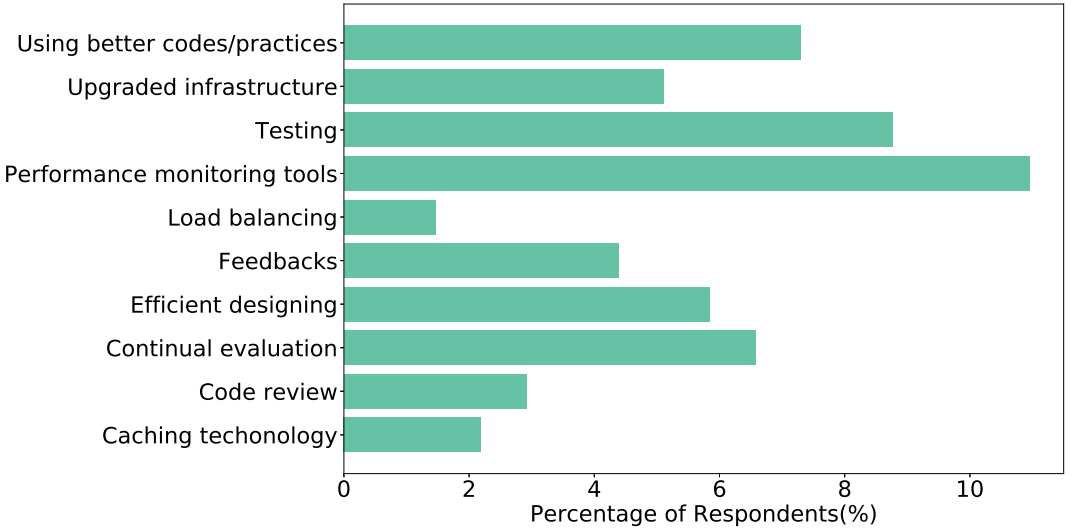


Fig. 18. Measures to ensure performance of products

#### 4.4.2 Performance.

Software performance indicates how efficient the software is in terms if resource and time. Testing tools such as load testing are common measure[?] to ensure performance of a software on this regard Bangladesh SE industry follows the same measures. According to our respondents the top three performance ensuring method in Bangladesh SE industry is use of tools, test and better codes/practices or design-implementation level measures.

- (1) **Caching Mechanism:** Caching mechanism improves performance by reducing response time.

*... Good Caching. [#29]*

- (2) **Emphasize on design:** Software performance is dependent on architecture of system. 5.84% respondents emphasize on design to ensure performance.

*By careful designing. [#24]*

- (3) **Testing:** Performance is ensured by various rigorous testing phase. The phase includes load testing, stress testing, integration testing.

*By rigorous testing and checking performance testing. [#17]*

- (4) **Emphasize on Infrastructure:** 5.11% respondents use upgraded infrastructure to ensure performance. These infrastructure includes cloud hosting, high end server, new technologies.

*Amazon Hosting and Quality Software. [#85]*

- (5) **Implementing best practices:** Industry standard best practices can improve system performance. 7.3% respondents ensure performance by implementing best practices. The best practices include compression technology, enforcing design patterns and refactoring.

*Implementation time carefulness and maintaining a well developed coding standard. [#40]*

- (6) **Load Balancing:** Load balancing can improve the system performance by ensuring equal load to all servers. 27.1% respondents think of using load balancing as a measure to maintain performance.

*Optimizing number of HTTP requests, Asynchronous programming, Caching, CDN, Load Balancing, nginx, varnish, compression of data, Continuous monitoring, Load testing, stress testing. [#42]*

- (7) **Performance Monitoring Tools:** There are various performance monitoring automated tools from which we can measure the overall as well as component-wise performance of a product. 10.95% respondents use these tools to measure performance.

*take help of different performance monitoring tools and dashboard, analyzed data , measure time and memory efficiency of process. [#35]*

- (8) **User Feedback:** It is a great source of performance measure. Taking a time to time feedback from the clients help a company to realise how their products are performing actually. Many (4.38%) respondents have highly recommended it.

*Continuous feedback from clients and QA team. [#65]*

- (9) **Continual Evaluation:** If there are some problems about performance, it can easily be identified in continuous monitoring and evaluation of the products. 6.57% of the total respondents have thought of this as a better solution.

*regular rechecking/running of automation scripts as a automation tester. [#14]*

- (10) **Code Review:** About 2.92% people have said that a proper and attentive code review can reduce the faults in the codes and therefore enhance the performance of a software.

*The code quality is assessed by the different team members during code review, followed by designing new ways to solve issues in the product that are time-intensive.. [#15]*

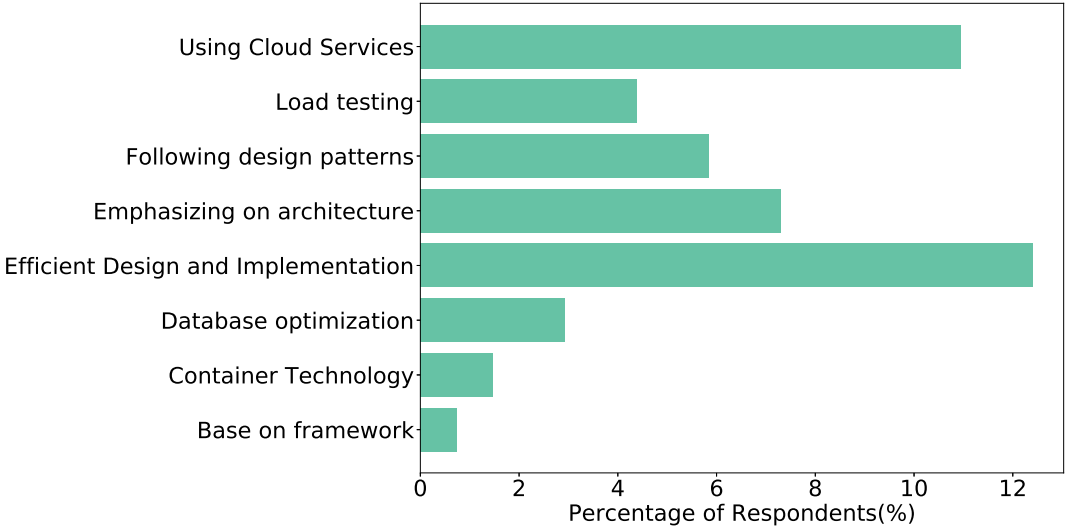


Fig. 19. Measures to ensure scalability of products

#### 4.4.3 Scalability.

Software scalability defines the ability to scale up a solution. Issues with little importance can impede scaling up. Thus proper measures should be taken from design stage to ensure the scalability of system. Recent days the cloud services offer tools to accomodate custom reactive scaling strategies. Thus, it has become easier to ensure scalability using cloud services [?]. Our results also follows the world trend, usage of cloud services has placed 2<sup>nd</sup> in terms popularity of scalability measures in SE industry of Bangladesh.

- (1) **Following Certain Design Patterns:** There are some design patterns which inherently help us in time of scaling. 5.84% people think that Following these design patterns will be of great use.

*Following certain design patterns. [#8]*

- (2) **Based on Framework:** Modern frameworks ensures scalability by default. 0.73% respondents solely depends on framework for scalability.

*following flexible framework which allows better scalability. [#14]*

- (3) **Container Technology:** Container technologies includes docker, Kubernetes which ensures OS level virtualization. By standardizing the system, container technologies eases the scaling of an infrastructure.

*We used Docker technology. [#85]*

- (4) **Database Optimization:** Database optimization includes sharding, clustering, indexing and scaling. 10.3% respondents optimize database to scale their system.



*Besides scaling horizontally, database scaling is performed by partitioning tables, along with multi-threaded implementations. [#85]*

- (5) **Efficient Design and Implementation:** 12.41% respondents emphasizes on the design and implementation of a scalable architecture.

*During implementation we always keep in mind about the scaling factor. [#40]*

- (6) **Emphasizing on architecture:** 7.3% respondents emphasized on architecture. It is mostly micro-service architecture which they use to ensure scalability.

*We follow the micro-service architecture. In a nutshell, we scale up the module vertically which is necessary. We use docker along with Jenkins for automatic deployments and scaling.. [#10]*

- (7) **Load Testing:** Load testing can be used to check the scalability of a system. 4.38% respondents use load test to check wheather their system is scalable or not.

*through load testing and load simulation.. [#35]*

- (8) **Using Cloud Services:** 10.95% user depends on Clod services like AWS and Azure for the scalability of system. Modern features like elastic load balancing and auto- scaling makes it easy to ensure scalability.

*Using AWS Elastic Load Balancer. [#28]*

#### 4.5 RQ5:What are the personal expectations and desire reigning in the industry?

This is the most interesting question. In fact, this question truly answers about the existing condition of our software industry. There are many problems and every one of them has solution. All about these have come into this question beautifully. Every employee has something to tell to their managers. Likewise, every manager has some demands from his subordinates. Not only these, they all have some expectations from government and also universities. All of these expectations are included in the answer of this research question.

- Expectation from Managers
- Expectation from Employees
- Expectation from Potential Candidates
- Expectation from Universities
- Expectation from Government
- Training provide by Companies

##### 4.5.1 Expectation from Managers.

- (1) **Proper behavioural manner:** 6.57% respondents expect well behaviour from management. It is clear from survey responses employees hates micro-management the most.

*Well Behaved and Proper Project Timeline. [#28]*

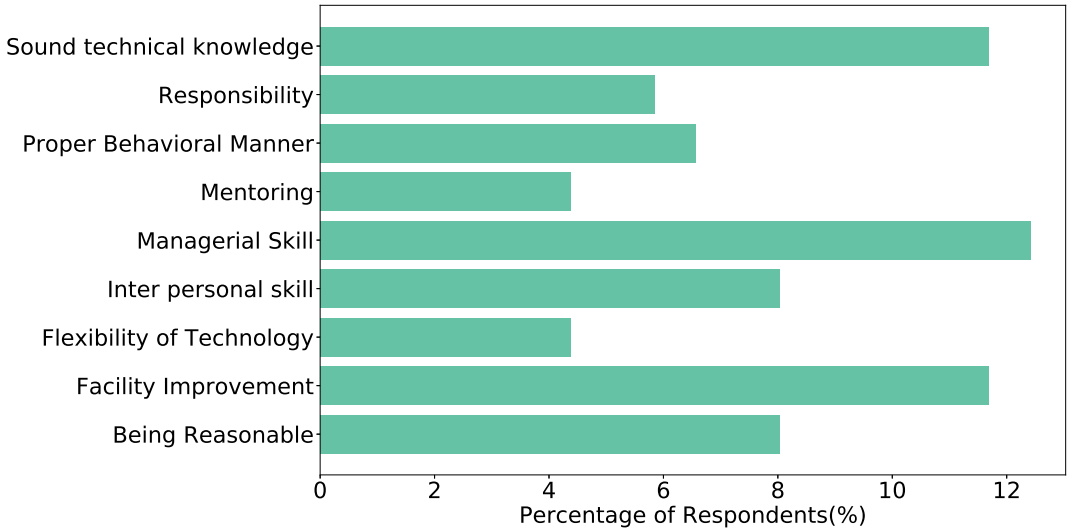


Fig. 20. Respondents expectation from managers

- (2) **Being Reasonable:** The managers should be reasonable in decision making. In order to get acquainted with the needs and necessities of the employees, managers should have a wide open view towards their desire. About 8.03% of the respondents have expressed in this way.

*Give reasonable time for a task and do proper comment on that task. [#38]*

- (3) **Flexibility of Technology:** It is very important for the managers to be up-to-date with the latest technology. Managers without a strong background of technical knowledge are really difficult to work with. 4.38% of total respondents have demanded for a manager with sufficient technical knowledge.

*Allow to use latest technologies and good practice to ensure company and their employee are updated with the current world. And also honor according to knowledge and quality works. In the company should have knowledge sharing, training sessions. [#48]*

- (4) **Responsibility:** Managers should take the responsibility for the works being done. A responsible manager can inspire the employees under him to a great extent. About 5.84% of the total respondents aspire for a responsible manager to work with.

*Take responsibility. [#82]*

- (5) **Facility Improvement:** 11.68% respondents mentioned that they look for facility improvements from managers. These facilities include increased remuneration, good working environments, team building events, positive recognition, yearly evaluation etc.

*Ensuring an environment where positive criticisms regarding code and people can be done..*

[#57]

- (6) **Inter Personal Skill:** 8.03% respondents expects inter personal skills among managers. The skills include proactive, leadership attitude, good communication skill and empathy.

*Ownership, Proactiveness, Leadership attitude, Good inter-personal and communication skills.*  
[#42]

- (7) **Managerial Skill:** The managers should have solid technical knowledge. He/she must be farsighted. One of the most important part of this post is management of requirements. He should have clear understanding of required manpower and resources. Respondents marked that timeline management is another managerial skill they expect from managers.

*1. Strong grasp on the requirements before assigning me to a project 2. Maintain informed and feasible deadlines.* [#8]

- (8) **Mentoring:** Respondents also look a mentor character in manager. They expect direction, advice and opportunity from managers for their career growth.

*To supervise and to advise to follow specific career plan.* [#9]

- (9) **Sound Technical Knowledge:** 11.68% respondents emphasizes on technical knowledge among managers.

*Should be technically active.* [#14]

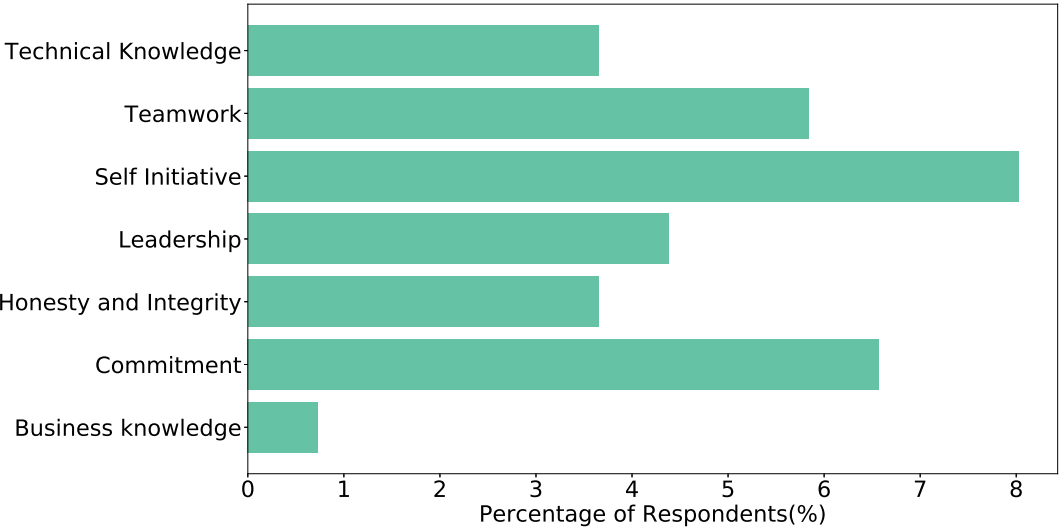


Fig. 21. Respondents expectation from employees

4.5.2 Expectation from Employees.

- (1) **Technical Knowledge:** 3.65% respondents look for solid technical knowledge and clean codes from employees.

*Code in a way that is a delight to read. [#49]*

- (2) **Teamwork:** It is very obvious to be expected. In a software industry, not a single product is created single-handedly. A bunch of people from various sectors work on the product, and to make a good product, team work between these members is very important. 5.84% of the managers responded desired this behaviour from the employees.

*Employees should be cooperative and share knowledge with each other and work together like family member. [#51]*

- (3) **Self Initiative:** An employee should learn in his own interest. In industry, everyone is occupied with himself. There is no space to be spoon-fed. A self-initiative worker will find his way even if there are many obstacles. 8.03% people have appraised self-initiative attitude.

*Clean code which creates less error. Coming up with optimization proposals on features developed earlier. Documentation in a timely manner. [#10]*

- (4) **Leadership:** 4.38% respondents look for leadership attitude like ownership of work, proactive and adaptiveness among employees.

*Sincere, Feel owner ship of project/product and good teamplayer. [#107]*

- (5) **Honesty and Integrity:** Not only coding and academic knowledge, but also behavioural manner plays a great role in deciding about a employee's career. 3.65% people want a well behaved and mannered employee to work with.

*honesty, integrity, self initiative and clear communication. [#35]*

- (6) **Commitment:** 6.57% managers have demanded commitment from their employees. A more committed employee will give better performance than a less committed employee, as he does not give full attention to the job.

*They will stay for long term and perform up to expectations, learn from mistakes. [#114]*

- (7) **Business Knowledge:** It is important to have knowledge of business domain when designing a solution or dealing with a customer. Thus respondents look for business knowledge like negotiation among employees.

*Understanding the business, end user and user stories. [#82]*

#### 4.5.3 Expectation from Potential Candidates.

- (1) **Team-player:** Team work plays the most significant role for success in a specific product in any industry. Every potential job candidate should be prepared to work together with a

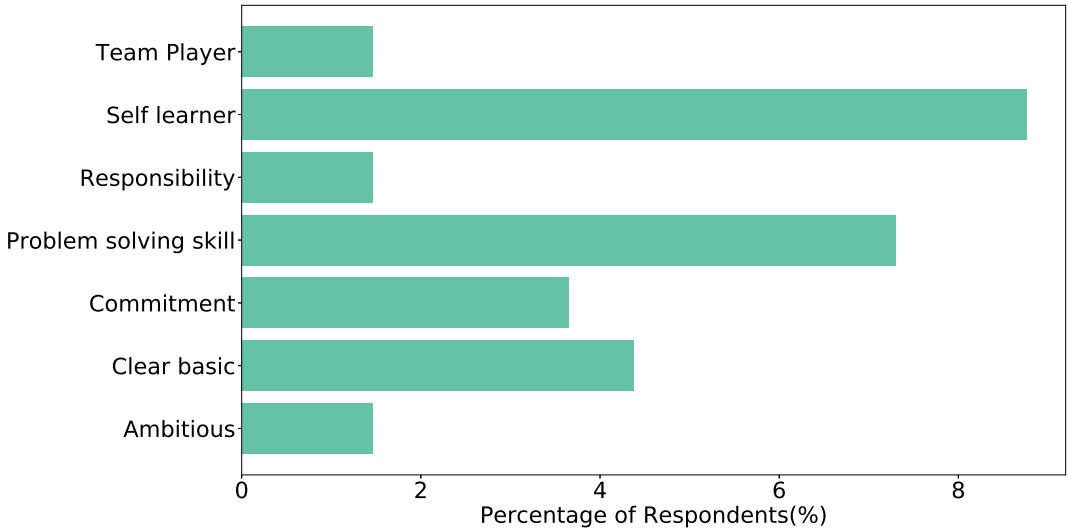


Fig. 22. Respondents expectation from potential candidates

group of people to fulfill the production of a software. 1.46% wanted to hire a team player than an individual talent.

*Enthusiastic, Self-learner, Team player. [#5]*

- (2) **Self-learner:** Most of the software companies does not have luxury to arrange training or to assign mentor. Thus employers look for self-learning capability among candidates.

*Self learner, Know well what you know. [#53]*

- (3) **Responsibility:** 1.46% respondents look for ownership of work or sense of responsibility among potential candidates.

*taking ownership of responsibility. [#66]*

- (4) **Problem-solving Skill:** Problem solving skill distinguishes a potential candidate from others. A candidate with good problem solving capability can adapt to any team. Problem-solving skill is desired by 7.3% of the total respondents.

*I am assuming that the potential candidates refer to the freshers in this question. I expect freshers to be very good with problem solving with strong analytical ability. Good university projects are plus. [#10]*

- (5) **Commitment:** 3.65% of the respondents want commitment from the freshers who are willing to enter into the software industry. He should give his best to attain the goal.

*Commitment and dedication.* [#114]

- (6) **Clear basic:** Every company want that freshers will have a clear understanding about the basic knowledge. They do not want knowledge about industry, they want clear knowledge about things like programming language, data structures and algorithm. 4.38% of the respondents want clear basics form the candidates.

*Professionalism and clear understanding of technologies what they know.* [#85]

- (7) **Ambitious:** 1.46% respondents prefer ambitious candidates. They look for enthusiasm and hunger for success among candidates.

*Knowing what he/she knows and what not, honesty, good problem solving capacities, team-man, absolutely no ego or arrogance, can get things done, ambitious, sincerity, discipline.* [#42]

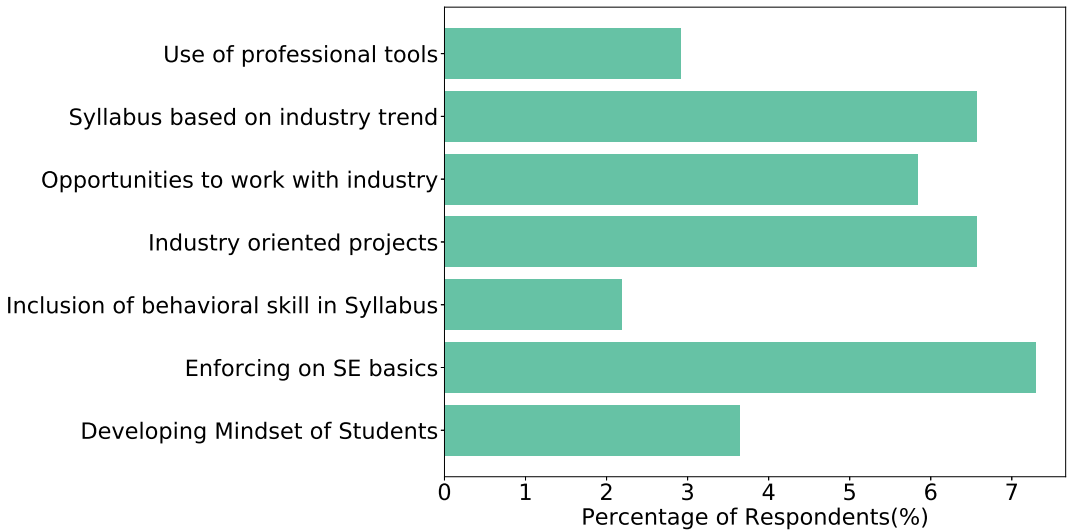


Fig. 23. Respondents expectation from university

#### 4.5.4 Expectation from University.

To have a vibrant SE industry and to maintain a steady workforce for the industry it is important that universities meet the expectation from the industry. As seen in Figure 23, industry owners/experts expect emphasis on SE basic and on current trends, they also seek for the opportunity to collaborate with the universities. The result coincides with a study conducted by Radermacher et al. [?]. According to their study, SE basic (Project Management, Software Tools, Testing) is the second area where graduate students are deficient in knowledge[?]. Though behavioral skill or soft skill enables faster integration and happier teams only 2.19% respondents emphasize on inclusion of behavioral skill in syllabus, the scenario is quite different in New Zealand. In a survey conducted by Stevens et al.[?] it is found that industry experts emphasize on behavioral skills after technical skill. They

also found that those skills are un-trainable in work place. Thus it is the duty of university to train graduates with required skill set.

- (1) **Inclusion of behavioral skill in Syllabus:** Being a good coder is difficult, but being a good human being is much more difficult. In industry, a person has to meet a large variety of personalities, and only a well behaved individual will find it easy to work with them. About 2.19% people have implied about it.

*Teach students Professional Ethics and Practices, Tie-up with Industry Practices and Problems, Solutions or Research Paper for any specific or generic Technical/Administrative problems. [#108]*

- (2) **Syllabus based on industry trend:** It is seen in some universities in our country that they are teaching on a syllabus which was designed a very very long time ago. Since then software industry has advanced a long way. 6.57% of our respondents think that these age-old academic curriculum should be changed and new latest topic should be added.

*Concentrate on applied science with theory. Change syllabus based on current trend each year. Apply internship to relevant organisations. [#5]*

- (3) **Opportunity to work with industry:** As many students will be joining the industry after graduation, it will be very convenient for everyone if they are acquainted with the industry in their varsity life. 5.84% respondents have felt the necessity of industry-academia correlation.

*Create opportunities to work with industry solving the problems we are facing in the industry. Sometimes we are so busy with developing features that we lack the opportunity to run some researches about our tasks. University projects can be subjected to s.... [#10]*

- (4) **Industry oriented projects:** 6.57% respondents want that in university student should engage in projects which is relevant to industry.

*Connection with industry, realistic teaching, internship arrangement for students. [#59]*

- (5) **Enforcing on SE basics:** 7.3% respondents emphasized on SE basics. They want prioritization of SE basic in university syllabus. The respondents mentioned design and delivery system, testing tools and design pattern in their response.

*University should train student some basic knowledge of industry practice. Research on industry SDLC.. [#115]*

- (6) **Developing mindset of students:** To have a keen interest to learn new technological knowledge is a must for every students. This mindset will not be developed in a day. They have to practise it for a long time. 3.65% respondents feel that teachers as well as universities can help them to achieve it.

*Develop mind set of student so that they love challenges, enjoy taking initiatives , never fear a problem and always ready to learn. It should teach students basic behavioral skills like*

*honesty, integrity, accept failure and criticism positively , respect... [#35]*

- (7) **Use of professional tools:** 2.92% respondents expects that university should make their student acquainted with professional software engineering tools like version controlling system, testing suits etc.

*They should introduce students to version controlling. Code quality can be also an important thing. [#57]*

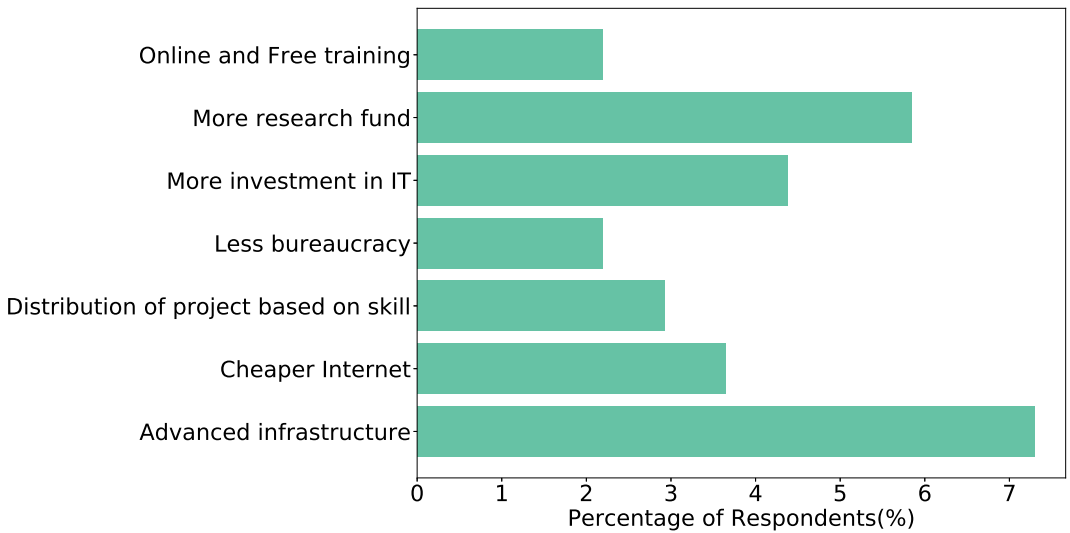


Fig. 24. Respondents expectation from government

#### 4.5.5 Expectation from Government.

Government policies shapes the future SE industry. Tessler et al.[?] completed a study to rank the possible action for government for boosting the IT industry. They have ranked infrastructure and telecommunication as the first two action for government. The result is similar to our study where the 7.3% respondents expects better infrastructure from government.

- (1) **Online and free training:** If government arranges free and online training, the professionals can learn a lot more alongside with the private training. 2.19% of our respondents have demanded for free online workshop.

*1.online Free training center for easy learning it from Advance level (industry level) , Not basic level. [#128]*

- (2) **More research fund:** In our country, research funds issued by government is really really less than expected. If we want to emerge as a country of developed software industry, government have to issue more money on research on this sector. 5.84% respondents think that it is very hard to develop software industry with such little funds.

*Issuing research funds to the universities to run researches. Government should emphasize in*



*automation in every possible sector. [#10]*

- (3) **More investment in IT:** 4.38% respondents want more investment in IT sector from the government. Respondents mention steps like tax waiver and creating opportunities for domestic companies in government projects.

*Tax benefits, infrastructure. [#66]*

- (4) **Less bureaucracy:** 2.19% of respondents want the removal of red tape from the administrative process as it prevents the development of the IT industry.

*get yourself out of corruption. and let yourself put in liability. [#133]*

- (5) **Distribution of project based on skill:** It is frequently seen in our country that less expert and less eligible persons are getting priority than more eligible persons because of nepotism. 2.92% of our respondents think that it should be eliminated as soon as possible for the sake of industry development.

*Proper maintenance with previous projects and follow up. Plan next years with long vision through solid IT-specialists. Distribute projects based on skills than links. Remove bureaucratic process for small companies to contribute more for country. [#5]*

- (6) **Cheaper internet:** In this age, it is difficult to spend a single day without internet. As it is also a great source of learning as well as earning, government should remove tax from internet charge and give internet in exchange of a very lower rate. Some people have also asked for 5G.

*faster and cheaper internet infrastructure. [#74]*

- (7) **Advanced infrastructure:** 7.3% respondents expect advanced infrastructure like IT/software parks, data center from government.

*Improved transport system, Software development zone with competitive rent. [#85]*

#### 4.5.6 Training provide by Companies.

In order to adapt to the new technologies and drive new employees, staff training sessions are required in SE firms. From the survey, it is clear that in-house training is the most popular form of training in the SE industry in Bangladesh. In the Australian [?] and Hong Kong [?] SE industry, the scenario is different where the most popular form of training is commercial courses and then in-house training. The difference in popularity can be linked to economic conditions. As the SE industry of Bangladesh is not mature, it is clear that SE companies will not be able to spend money on paid training courses.

- (1) **Using paid resources:** 4.38% respondents mentioned that they use online paid resources or certification courses to train their employees.

*We have pluralsight account and we perform bimonthly technical session. [#57]*

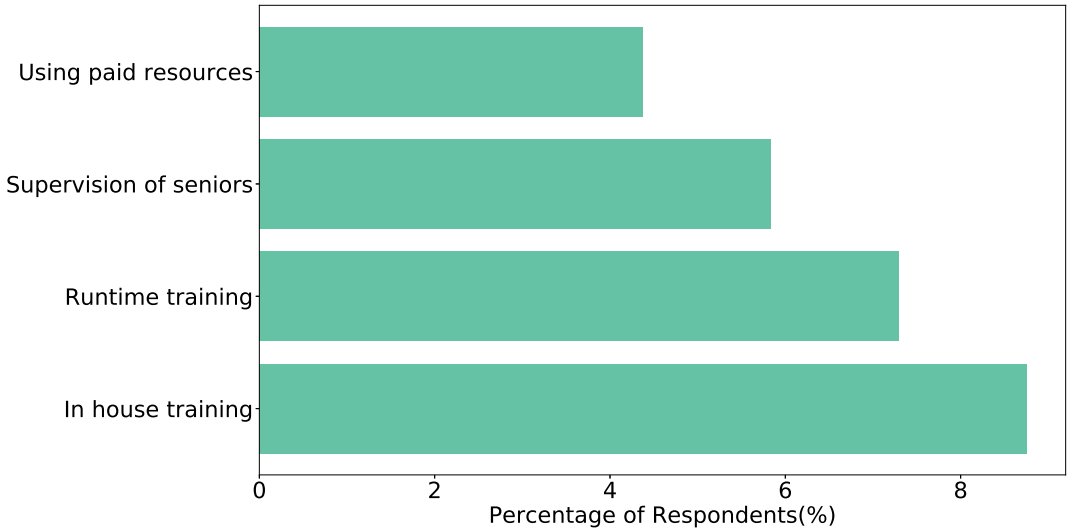


Fig. 25. Respondents expectation from government

- (2) **Supervision of seniors:** Being supervised by the seniors is the best way of training. They can teach us from their experience and experience teaches a lot.

*mentoring.* [#74]

- (3) **Runtime training:** About 7.3% people have said that they provide their employees with runtime training.

*Run time training through small scale project and supervision of seniors.* [#5]

- (4) **In house training:** Companies also provide their employees with tutorials and books so that they can read at their home and get used to the practices followed.

*Internal knowledge sharing sessions.* [#53]

## 5 CONCLUSION

Thanks to the results of this survey, we have observed that software industry in Bangladesh is spirited, growing and competitive that many software teams are eager for adopting state-of-the-art software engineering practices and approaches. Awareness and knowledge about these practices and approaches are rising in the industry, that many organizations have been establishing them within their development operations. We have also observed that the industry needs to improve on using well-established requirements engineering and testing related practices. These two areas are where most displeasure is stated about. Also we observed that practitioners have a tendency to avoid formal or relatively complex best practices (e.g. TDD, formal languages, pair programming, automation testing and metric usage) in general, but especially in these two areas. We suggest organizations to invest in improvement and training efforts for increasing the awareness and

knowledge about these software disciplines. In this study, we have seen that social aspects are important as well that many practitioners have expressed their discontent and difficulties on issues like communication, collaboration, and team structure.